数理解析研究所講究録
第 695 巻 1989 年 145-154

Indexing Functions and Time Lower Bounds for Sorting on a Mesh-Connected Computer

Yijie Han†, Yoshihide Igartashi†,‡,
韓以捷, 五十嵐善英
Miroslaw Truszczynski†

† Department of Computer Science, University of Kentucky,
Lexington, Kentucky 40506, U.S.A.
‡ Department of Computer Science, Gunma University,
Kiryu, 376 JAPAN

We introduce a parameter of indexing functions and show its relation to lower bounds for sorting algorithms on mesh-connected computers that follow from the Chain Theorem. We give lower and upper bounds for the parameter. Conclusions from our results are: (1) no matter what indexing function is used, any sorting algorithm must execute $2.27n + \Theta(1)$ steps; (2) the best lower bound true for all indexing functions that we can hope to prove by the Chain Theorem argument is $2.5n + \Theta(1)$.

1 Introduction

In this paper we study a combinatorial problem that arises in considerations of sorting problems on a mesh-connected computer. As usual in such cases, it is of main concern to design fast algorithms and to prove lower bounds for the complexity of the problem to get an idea how good designed algorithms are.

Sorting on a mesh-connected computer has received much attention lately ([HI1,HI2, K1,K2,MSS, SS]). It turns out that the efficiency of a sorting algorithm depends on the indexing function used (see [HI1]). For a snake-like row-major indexing scheme an algorithm running in $3n + o(n)$ steps is known (Schnorr and Shamir [SS]), and it is also known to be optimal (Kunde [K1], Schnorr and Shamir [SS]). So far, no sorting algorithm is known that would run in $(3 - \varepsilon)n + o(n)$ steps, for some $\varepsilon > 0$. Also, the snake-like row-major is the only (up to trivial variations) indexing schemes for which fastest algorithms are known to be optimal.

The element whose final location is in the processors in a corner of the mesh may be initially stored in the processor in the opposite corner, and it takes at least $2n$ steps merely to move it to its proper final destination. This "structure based" lower bound is too weak. Only recently, a more powerful lower bound technique, known as *joker-zone* method, was discovered by Kunde [K1] and Schnorr and Shamir [SS]. Their method was subsequently refined by Han and Igarashi [H1]. They developed an argument based on the so called *Chain Theorem*, and proved that $(1+\sqrt{6}/2)n+\Theta(1)$ is a lower bound for the running time of any sorting algorithm, no matter what indexing function is used.

The main goal of this paper is to study the power of the Chain Theorem of [HI1] in proving lower bounds for sorting algorithms. To this end, for an indexing function I we define a combinatorial parameter called *stretch* $s(I)$, and show that lower bounds implied by the Chain Theorem directly depend on this parameter. Our first main result provides a lower bound for $s(I)$; this allows to prove that independent of an indexing function, every sorting algorithm requires at least $2.27n$ steps, an improvement over the old bound of $(1+\sqrt{6}/2)n+\Theta(1)$ of [HI1]. Our second result exhibits an indexing function I with $s(I)=0.5n+\Theta(1)$.

2 Preliminaries and problem formulation

We consider a general model of a synchronous $n\times n$ mesh-connected processor array as given in [SS]. It is denoted by $M[0..m,0..m]$; here, and throughout the paper $m=n-1$. Each processor at location $(i,j), 0\le i,j\le m$, is denoted by $M[i,j]$. The distance between $M[i_1,i_2]$ and $M[j_1,j_2]$ is defined as $|i_1-j_1|+|i_2-j_2|$ and denoted by $d((i_1,i_2),(j_1,j_2))$. Processor $M[i_1,i_2]$ is directly connected with processor $M[j_1,j_2]$ if and only if $d((i_1,i_2),(j_1,j_2))=1$. All n^2 processors work in parallel with a single clock, but they may run different programs. As for sorting computation, the initial contents of $M[0..m,0..m]$ are assumed to be n^2 items drawn from a totally ordered set, where each processor has exactly one item. The final contents of $M[0..m,0..m]$ is the sorted sequence of the items in a specific order. In one step each processor can communicate with all of its directly connected neighbor processors. The interchange of items in a pair of directly connected processors or the replacement of the item in a processor with the item in one of its directly connected processors can be done in one step. The computing time is defined as the number of parallel steps of such basic operations to reach the final configuration.

A one-to-one function $I : \{0, 1, ..., m\}^2 \rightarrow \{1, 2, ..., n^2\}$ is called an *indexing function.* Given an indexing function I, the goal is to sort n^2 items initially stored in the n^2 processors so that when the algorithm terminates, the item of rank k (the *k-th* smallest) is located in processor $M[i, j]$, where $I(i, j) = k$.

A subset of $M[0..m, 0..m]$ is called a *region.* For a region S the number of processors in S will be called the *cardinality* of S and will be denoted $|S|$. A set $\{(i_{11}, i_{12}), ..., (i_{c1}, i_{c2})\}$ is called a *chain under indexing function I* (or a *chain* if I is understood) if $\{I(i_{11}, i_{12}), ..., I(i_{c1}, i_{c2})\}$ is a set of consecutive integers. The *length* of such a chain is c. If (i_1, i_2) is in $\{0, m\}^2$ and x is a positive real number, $\{M[j_1, j_2] : d((i_1, i_2), (j_1, j_2)) \leq x\}$ is called a *corner* region and is denoted by $CREG((i_1, i_2); x)$. An *open* corner region is the set $\{M[j_1, j_2] : d((i_1, i_2), (j_1, j_2)) < x\}$, and it is denoted by $CREG_o((i_1, i_2); x)$. The set of all processors that are at distance at least $m - x$ from all four processors $M[0, 0], M[0, m], M[m, 0]$, and $M[m, m]$ is called a *center* region and is denoted by $CENT(x)$. Throughout the paper, for any two real numbers a and b, $[a, b]$ denotes the set of all integers j, such that $a \leq j \leq b$.

Consider now an indexing function I and a corner region $R = CREG_o((i, j); x)$, for some real x, $0 \leq x \leq 2m$. Let c be the length of a longest chain contained in R, and let $t(R)$ be the smallest real number t such that $c \leq |CREG((i, j); t)|$. Finally, put $s(R) = x - t(R)$. The *stretch* $s(I)$ of I is defined as $s(I) = sup\ s(R)$, where the supremum is taken over all corner regions R. The next theorem has been derived in [HI1] and is called the Chain Theorem.

Theorem 2.1 (Chain Theorem [HI1]) *Let I be an indexing function. Then, every algorithm for sorting n^2 items into the order specified by I takes at least $2n + s(I) + \Theta(1)$ steps.*

In this paper we study the parameter $s_n = min\ s(I)$, where the minimum is taken over all possible indexing functions on an $n \times n$ mesh of processors. We show that $0.27n \le s_n$(hence, every sorting algorithm must require at least $2.27n$ steps), and that $s_n \le 0.5n$ (hence, the best universal lower bound that can be obtained using the Chain Theorem only is $2.5n$).

3 Lower bounds

In this section we will show two theorems each giving a lower bound for s_n. The first one gives the lower bound initially presented in [HI1]. We present here a different proof of that result which is simpler than the original one and helps better understand the approach behind the proof of the improved lower bound for s_n (Theorem 3.3).

Lemma 3.1 *Let a be a real number, $0 \le a \le 1/2$.*

(a) $I^{-1}([an^2, (1-a)n^2]) \subseteq CENT((1-\sqrt{2a})n + s(I) + \Theta(1))$.

(b) Let $x_a = inf\{x : |CENT(x)| \ge |[an^2, (1-a)n^2]|\}$.*We have*

$$x_a = \begin{cases} n\sqrt{1/2 - a} + \Theta(1) & if \;\; 1/4 < a \le 1/2 \\ n(1-\sqrt{a}) + \Theta(1) & if \;\; 0 \le a \le 1/4 \end{cases}$$

Proof. (a) Let b be an integer, $b \in [an^2, (1-a)n^2]$. Suppose $I^{-1}(b) \notin CENT((1-\sqrt{2a})n + s(I) + 1)$. Then, for some $(i,j) \in \{0, m\}^2$, $I^{-1}(b) \in CREG_0((i,j); n\sqrt{2a} - s(I) - 2)$. Let $d((i,j), I^{-1}(b)) = x$ and let $R = CREG((i,j), 2m - x)$. Clearly, $x < n\sqrt{2a} - s(I) - 2$, and the longest chain contained in R has length at most $(1-a)n^2$. Hence, $t(R) \le 2n - n\sqrt{2a}$. Consequently, $s(R) = 2m - x - t(R) > s(I)$, a contradiction.
(b) Follows directly from the following formula for the number of elements in a center region.

$$|CENT(x)| \le \begin{cases} 2x^2 + \Theta(x) & if \;\; 0 \le x \le m/2 \\ n^2 - 2(m-x)^2 + \Theta(m-x) & if \;\; m/2 < x \le m \end{cases} \qquad \square$$

Theorem 3.1 (Han and Igarashi, [HI1]) $s_n \ge (\sqrt{6}/2 - 1)n + \Theta(1)$.

Proof. Let us consider an arbitary indexing function I. Under the notation from

Lemma 3.1 we have

$$x_a \leq n + s(I) - \sqrt{2a}\,n + \Theta(1)$$

(this follows from Lemma 3.1 (a)). Hence (by Lemma 3.1 (b)),

$$s(I) \geq \begin{cases} n(\sqrt{1/2-a}+\sqrt{2a}-1)+\Theta(1) & if \;\; 1/4 < a \leq 1/2 \\ n(\sqrt{2a}-\sqrt{a})+\Theta(1) & if \;\; 0 \leq a \leq 1/4 \end{cases}$$

Maximizing the right hand side with respect to a we get $s(I) \geq (\sqrt{6}/2 - 1)n + \Theta(1)$, as claimed. □

Next, we present an improvement on this result. We first prove an auxiliary lemma.

Lemma 3.2 *Consider two center regions B_1 and B_2 and regions C_i and D_i, $i = 1,2,3,4$ as shown in Fig. 3.1. Define $H_i = C_i \cup D_i \cup C_{i+1}$, $i = 1,2,3$, and $H_4 = C_1 \cup D_4 \cup C_4$. Put $B = B_1 - B_2$, $b = |B|$, and $d = |D_1|$ (regions D_i have all the same size). Assume that more than $b/2 + 2d$ elements of B are colored with blue and green and suppose that there is at least one element of each color. Then at least one of the regions H_i contains elements of both colors.*

Proof. Without loss of generality we may assume that there are no less blue elements than green elements.

Case 1. There exists a green element in $\cup_{i=1}^{4} C_i$. Without loss of generality we may assume that there is a green element in C_1. Since more than $b/4 + d$ elements are colored blue, there exists a blue element not in $C_3 \cup D_2 \cup D_3$, and the assertion of the lemma holds.

Case 2. All green elements are in $\cup_{i=1}^{4} D_i$. Without loss of generality we may assume that there is a green point in D_1. Since more than $b/2 + 2d$ elements in B are colored and no green element is in $C_1 \cup C_2$, there is at least one blue element in $C_1 \cup C_2$. Thus, the assertion of the lemma holds in this case, too. □

Theorem 3.2 $s_n \geq 0.27n + \Theta(1)$.

Proof. Let I be an arbitary indexing function. Suppose that $s(I) < 0.27n$. Consider two sets $A_1 = I^{-1}([0.21n^2, 0.79n^2])$ and $A_2 = I^{-1}([0.395n^2, 0.605n^2])$. By Lemma 3.1(a),

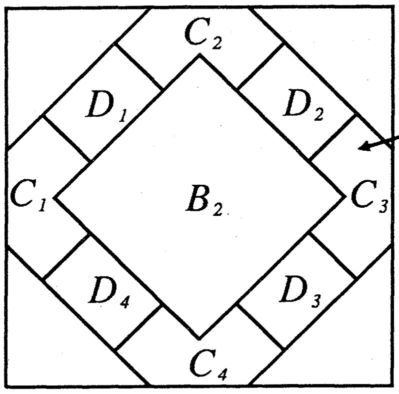

Figure 3. 1 Regions on the mesh-connected model

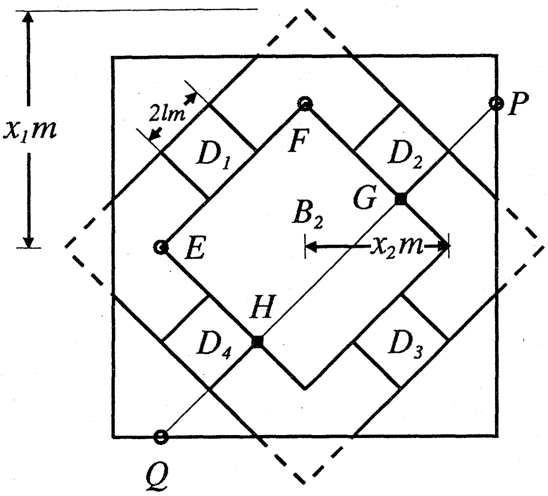

Figure 3. 2 Regions for the proof of Theorem 3. 2

for every sufficiently large n, $A_i \subseteq B_i$, where $B_i = CENT(x_i m)$, $x_1 = 0.622$ and $x_2 = 0.3812$ (recall that $m = n - 1$). Regions B_i and other regions we will consider in the proof are shown in Fig. 3.2. Let $B = B_1 - B_2$ and $l = 0.1123024$. Color all elements in $I^{-1}([0.21n^2, 0.395n^2])$ in blue and all elements in $I^{-1}([0.605n^2, 0.79n^2])$ in green. Altogether, there are $0.37n^2 + \Theta(1)$ colored elements. These colored elements must be located in B_1. At most $|B_2| - (0.21n^2) + \Theta(1)$ of them can be located in B_2, as $A_2 \subseteq B_2$ and $|A_2| = 0.21n^2 + \Theta(1)$. Since $|B_2| = 2(x_2m)^2 + \Theta(m) = 2(x_2m)^2 + \Theta(m)$, at least $0.2893m^2 + \Theta(m)$ of the colored elements are located in B. In particular, it follows that B contains both blue and green points. Observe that $|B| = 2(x_1m)^2 - 2(x_2m)^2 - 4(x_1m - m/2)^2 + \Theta(m)$. Denote by d the common cardinality of regions D_i and observe that $d = \sqrt{2}l(x_1 - x_2)m^2 + \Theta(m)$. Hence, the number of colored elements in B is bigger (for sufficiently large n) than $|B|/2 + 2d$. Thus, by Lemma 3.3, there are both blue and green points in one of the regions H_i (see notation of Lemma 3.3), say in H_3. Consider now set $A_2 = I^{-1}([0.395n^2, 0.605n^2])$. It has $0.21n^2 + \Theta(1)$ elements. All of them belong to B_2. Notice, that the cardinality of the region $EFGH$ is given by $(x_2m)^2 + \sqrt{2}lx_2m^2 + \Theta(m)$ and thus it contains at most $0.206m^2 + \Theta(m)$ elements. Therefore, for every sufficiently large n, there is an element in A_2 that belongs to B_2-$EFGH$. Let R be the corner region determined by the line PQ and containing H_1. It follows that the longest chain in R has length at most $0.395n^2 + \Theta(1)$. Thus,

$s(R) \geq 0.27n + \Theta(1)$, as required. □

We conclude this section with a theorem being a corollary of Theorems 2.1 and 3.2.

Theorem 3.3 *No matter what indexing function is used, any algorithm for sorting n^2 items on a mesh-connected computer takes at least $2.27n + \Theta(1)$ steps.*

4 Limit of the chain argument

In this section we study the power of the chain argument. It turns out that the best lower bound we can hope to obtain using this type of an argument is $2.5n + \Theta(1)$. To justify this claim we will construct an indexing function I with $s(I) \leq 0.5n + \Theta(1)$.

We denote $CREG_0((i,j), \lceil m/2 \rceil)$, for $i, j \in \{0, m\}^2$ by $A_{i,j}$ and $CENT(\lceil m/2 \rceil)$ by C. Let $a = |A_{i,j}|$ (it does not depend on i and j) and $c = |C|$. Let us assume now that an indexing function I satisfies the following requirements:

(1) Processors in $A_{0,0}$ (resp. $A_{0,m}$) will be assigned odd (resp. even) integers from $\{1, ..., 2a\}$, processors in C will be assigned elements from $\{2a+1, 2a+2, ..., n^2 - 2a\}$, and processors in $A_{m,0}$ (resp. $A_{m,m}$) will be assigned odd (resp. even) integers from $\{n^2 - 2a + 1, n^2 - 2a + 2, ..., n^2\}$.

(2) For every $x = M[i_1, j_1]$ and $y = M[i_2, j_2]$,

  (a) If x and y are both in $A_{0,0}$ or in $A_{m,m}$ and $i_1 - j_1 < i_2 - j_2$, then $I(x) > I(y)$.

  (b) If x and y are both in $A_{0,m}$ or in $A_{m,0}$ and $i_1 + j_1 < i_2 + j_2$, then $I(x) > I(y)$.

  (c) If x and y are both in C and $i_1 < i_2$, then $I(x) < I(y)$.

An expample of an indexing function satisfying requirements (1) and (2) (for $n = 9$) is given in Fig. 4.1. It is clear that indexing functions satisfying (1) and (2) exist for every positive n.

Theorem 4.1 *If an indexing function satisfies requirements (1) and (2), then $s(I) = 0.5n + \Theta(1)$. Hence, $s_n \leq 0.5n + \Theta(1)$.*

Proof. To prove the theorem, we show that no matter what corner region $R =$

$CREG((i,j);x)$ is used, $s(R) = x - t(R) \leq 0.5n + \Theta(1)$. We consider first the case 1, 2, 3 and 4, the elements contained in the interior of region B indicated with the bold line in Fig. 4.2(a), (b) and (c), respectively, form a chain. (In this figure, we assume that top leftmost corner contains $M[0,0]$, and top rightmost corner contains processor $M[0,m]$.)

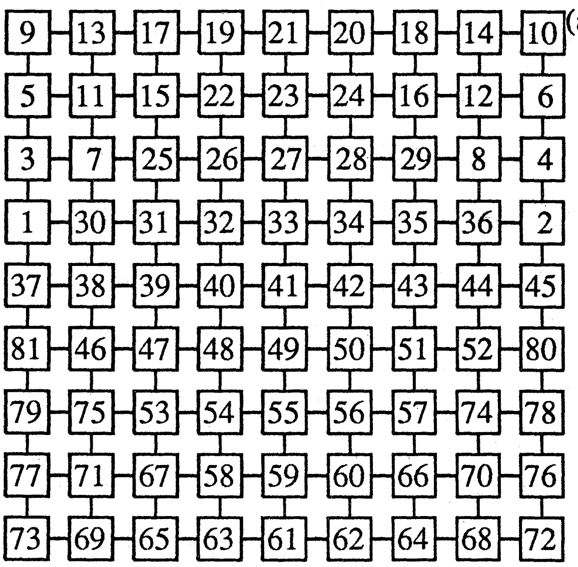

Figure 4.1 An indexing scheme

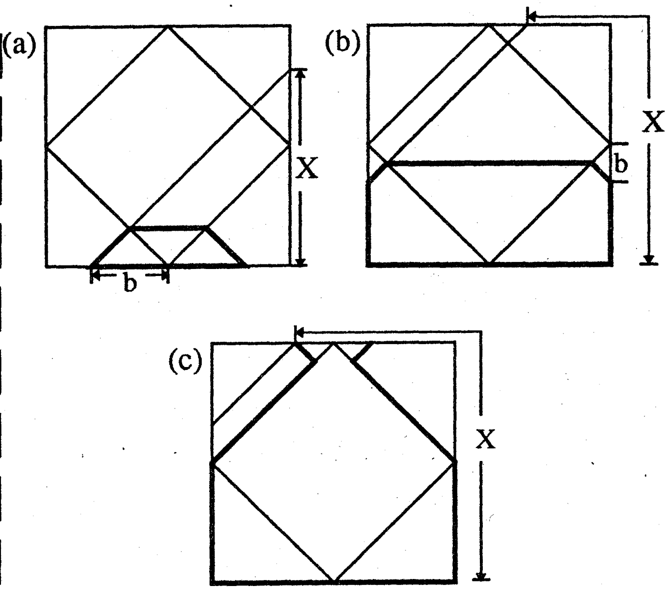

Figure 4.2 Chains formed in regions

The length of this chain is equal to $|B| + \Theta(p)$, where $|B|$ is the area of the polygon B and p is the perimeter of B. (Note that in (c) B is the union ot two interior-disjoint convex polygons.)

1. $0 \leq x \leq 0.5m$. In this case $s(R) \leq x \leq 0.5m$, as required.

2. $0.5m < x \leq m$. In this case, (see Fig. 4.2(a)) $|B| = 3b^2/4$ and $p = \Theta(b)$, where $b = x - 0.5m$. Hence, $t(R) = b\sqrt{3/2} + \Theta(1)$. Since $0 < b \leq 0.5m$, $s(R) = x - t(R) \leq 0.5m + \Theta(1)$.

3. $m < x \leq 1.5m$. In this case, (see Fig 4.2(b)) $|B| = (2m^2 - 2bm - b^2)/4$ and $p = \Theta(m)$, where $b = 1.5m - x$. Since $0.1875m^2 + \Theta(m) \leq |B| \leq 0.5m^2$, $s(R) = 1.5m - b - \sqrt{(2m^2 - 2bm - b^2)/2}$
$+ \Theta(1)$. As $0 \leq b < 0.5m$, also in this case we have $s(R) \leq 0.5m + \Theta(1)$.

4. $1.5m < x \leq 2m$. $|B| = 0.75m^2 + (x - 1.5m)^2/2$ and $p = \Theta(m)$. $|B| \geq 0.75m^2 +$

$\Theta(m)$ so, $s(R) = \sqrt{0.5m^2 - (x - 1.5m)^2} + x - 2m + \Theta(1)$. Since, $1.5m < x \leq 2m$, $s(R) \leq 0.5m + \Theta(1)$ follows.

In the other three cases for the corner we consider identical subcases and get the same formulas for $t(R)$ and $s(R)$ as in the corresponding subcase for the corner (m, m). This completes the proof. □

5 Concluding remarks

We showed that $0.27n + \Theta(1) \leq s_n \leq 0.5n + \Theta(1)$. Improving the lower bound on s_n would give a better lower bound for sorting on a mesh-connected computer. The upper bound on s_n, which we proved by exhibiting a class of indexing functions I with $s(I) = 0.5n + \Theta(1)$ shows the limit of the Chain Theorem in proving lower bounds. Although the Chain Theorem is strong enough to prove optimality (up to the leading term) of the $3n + o(n)$ algorithm of Schnorr and Schamir [SS], it seems unlikely that an indexing function exists that would admit a sorting algorithm running in $(3 - \varepsilon) + o(n)$ steps. So, in general, stronger lower bound techniques are needed.

Recently Kunde also showed a lower bound of $0.25n$ on s_n [K3]. However, our lower bound of $0.27n$ on s_n is still the best one known. Using a more complicated indexing scheme we can show an upper bound of $0.46n$ on s_n [HIT2].

References

[HI1] . Y. Han and Y. Igarashi, Time lower bounds for sorting on a mesh-connected processor array, in Proceedings of Aegean Workshop on Computing, 1988, Lecture Notes in Computer Science, Vol.319, Springer-Verlag, pp.434-443.

[HI2] . Y. Han and Y. Igarashi, Time lower bounds for parallel sorting on multidimensional mesh-connected processor arrays, in Proceedings of International Conference on Parallel Processing, 1988, pp.194-197.

[HIT1]. Y. Han, Y. Igarashi, M. Truszczynski, Indexing functions and time lower bounds on a mesh-connected computer, Technical Report No.114-88, Dept, of Computer Science, University of Kentucky, May 1988.

[HIT2]. Y. Han, Y, Igarashi, M. Truszczynski, Technical Memo, Dept. of Computer Science, University of Kentucky, 1988.

[K1] . M. Kunde, Lower bounds for sorting on mesh-connected architectures, Acta Informatica, 24 1987, pp.121-130.

[K2] . M. Kunde, Optimal sorting on multi-dimensionally mesh-connected computers, 4th Symp. on Theoretical Aspects of Computer Science, LNCS 247 Springer, 1987, pp.408-419.

[K3] . M. Kunde, Bounds for l-selection and related problems on grids of processors, Technical Report, Institute für Informatik. Technishe Universität, 1988 (Proceedings of Paralell 88, Berlin, EDR 1988)

[MSS]. Y. Ma, S. Sen and I.D. Scherson, The distance bound for sorting on mesh-connected processor arrays is tight, 27th Symp. on Foundations of Comput. Sci., IEEE, 1986, pp.255-263.

[SS] . C. P. Schnorr and A. Shamir, An optimal sorting algorithm for mesh-connected computers, Proc. 18-th ACM Symp. on Theory of Computing, 1986, pp.255-263.